

## Viterbi decoder

Patent Number: ☐ US5982822  
Publication date: 1999-11-09  
Inventor(s): HATAKEYAMA IZUMI [JP]  
Applicant(s): SONY CORP [JP]  
Requested Patent: ☐ JP9232972  
Application Number: US19970798548 19970211  
Priority Number(s): JP19960041352 19960228  
IPC Classification: H03D1/00; H04L27/06; G06F11/10  
EC Classification:  
Equivalents:

---

### Abstract

A viterbi decoder that performs viterbi decoding at a high speed with low power consumption using a memory such as a random access memory as a path selection information storage unit. Adder-comparator-selector (ACS) processing units 13a, 13b time-divisionally perform ACS processing for even and odd number states, respectively, based on the branch metrics from a branch metric computing unit 12 and the path metric of the previous period to produce the path selection information. Eight states of the path selection information storage are collected together by a temporary storage circuit 20 and sent to and written in a path selection information storage unit 15. In the path selection information storage unit 15, the past path selection information is read out during the time other than a write time for the path selection information. The path selection information corresponding to the maximum likelihood states of the path selection information in the path selection information storage unit 15 is read out during the time the ACS processing is performed for all states.

---

Data supplied from the esp@cenet database - I2

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-232972

(43) 公開日 平成9年(1997)9月5日

(51) Int.Cl.<sup>8</sup>

H 0 3 M 13/12

識別記号

庁内整理番号

F I

H 0 3 M 13/12

技術表示箇所

審査請求 未請求 請求項の数 2 O L (全 9 頁)

(21) 出願番号

特願平8-41352

(22) 出願日

平成8年(1996)2月28日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 畠山 泉

東京都品川区北品川6丁目7番35号 ソニー株式会社内

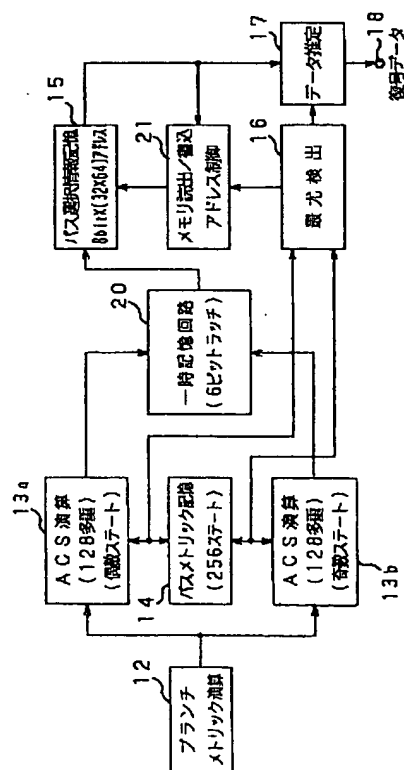
(74) 代理人 弁理士 小池 晃 (外2名)

(54) 【発明の名称】 ビタビ復号器

(57) 【要約】

【課題】 ビタビ復号器のパス選択情報記憶部にRAM等のメモリを用いながらビタビ復号の高速化及び低消費電力化を図る。

【解決手段】 ブランチメトリック演算部12からのブランチメトリックと前時刻のパスメトリックとに基づいて、ACS演算部13a、13bで、偶数、奇数ステートのACS演算をそれぞれ時分割処理により行い、パス選択情報を得て、一時記憶回路20により8ステート分のパス選択情報をまとめ、パス選択情報記憶部15に送って書き込んでいる。パス選択情報記憶部15では、上記8ステート分のACS演算処理の間に、パス選択情報の書き込み以外の時間で過去のパス選択情報の読み出しを行い、全ステートのACS演算を行う間に、パス選択情報記憶部15内の全ての過去についての最尤ステートに対応するパス選択情報を読み出す。



パスメトリックメモリ近傍の構成例を示す図

## 【特許請求の範囲】

【請求項1】 入力符号に基づいてブランチメトリックを計算するブランチメトリック計算手段と、

このブランチメトリック計算手段からのブランチメトリックに基づいて各ステートのパスメトリック及びパス選択情報を時分割処理により計算するACS演算手段と、このACS演算手段からのパス選択情報を複数のnステート分毎にまとめて記憶するパス選択情報記憶手段とを有し、

上記nステート分のACS演算処理中の上記パス選択情報記憶手段へのパス選択情報の書き込み時以外の時間に上記パス選択情報記憶手段から最尤判定のための過去のパス選択情報を読み出すことを特徴とするビタビ復号器。

【請求項2】 上記ACS演算されて得られたパス選択情報を一時記憶して上記nステート分をまとめて上記パス選択情報記憶手段に送る一時記憶手段を設けることを特徴とする請求項1記載のビタビ復号器。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】本発明は、畳み込み符号の最尤復号に用いられるビタビ復号器に関するものである。

## 【0002】

【従来の技術】ビタビ (Viterbi) 復号器は、畳み込み符号の最尤復号法に使用されるものであり、既知の複数の符号系列のうち、入力符号系列に最も符号距離に近いパスを最尤パスとして選択し、この選択されたパスに対応して復号データを得るものである。ビタビ復号は、通話路に生じるランダム誤りに対する訂正能力が高く、軟判定復調方式と組み合わせると、特に大きな符号化利得を得ることができる。例えば、干渉波の影響を受け易く、電力制限の厳しい衛星通信システムでは、誤り訂正符号として畳み込み符号が用いられており、その復号にビタビ復号器が用いられている。

【0003】このビタビ復号アルゴリズムについて簡単に説明する。例えば生成多項式が

$$G_0(D) = 1 + D^2$$

$$G_1(D) = 1 + D + D^2$$

で与えられる符号化率 $R = 1/2$ 、拘束長 $K = 3$ の畳み込み符号を考える。このような符号を発生する符号器の構成例を図6に示す。この図6において、入力データは2個のフリップフロップ等の遅延素子301、302により順次遅延され、加算器303により入力データと遅延素子302からのデータが加算されて出力 $G_0$ として取り出され、加算器304、305により入力データと遅延素子301、302からのデータとが加算されて出力 $G_1$ として取り出される。

【0004】このような符号器における各遅延素子301、302からの各出力をそれぞれ $b_1$ 、 $b_2$ とすると、符号器の状態( $b_2$   $b_1$ )としては、(00)、

(01)、(10)、(11)の4つの状態が取り得る。また、入力を与えられたときの遷移し得る状態は常に2通りとなる。

【0005】すなわち、入力0のとき、現在の状態が(00)又は(10)の場合に状態(00)に遷移し、現在の状態が(01)又は(11)の場合に状態(10)に遷移する。入力1のときには、現在の状態が(00)又は(10)の場合に状態(01)に遷移し、現在の状態が(01)又は(11)の場合に状態(11)に遷移する。

【0006】このような状態遷移をトレリス線図で示すと図7ようになる。この図7において、実線のブランチ(枝)は入力0のときの遷移を示し、破線のブランチは入力1のときの遷移を示す。また、各ブランチに沿って記されている数字は、そのブランチの遷移が生じたときに出力される上記図6の符号( $G_0$   $G_1$ )である。

【0007】この図7から明らかなように、各状態への遷移時には、必ず2つのパスが合流する。ビタビ復号アルゴリズムにおいては、各状態への2つのパスの内、最尤(最も確からしい)パスを選択し、所定長まで生き残りパスの選択を行って、各状態で選択したパスの内の最尤のものを検出することで、受信符号を復号する。

【0008】このようなビタビアルゴリズムに基づいて畳み込み符号を復号するビタビ復号器は、基本的に送信系列と各ブランチとの間のメトリックを計算するブランチメトリック計算手段と、生き残りパスを選択して生き残りパスのパスメトリックを計算するACS(アダー、コンパレータ、セレクト)演算手段と、各ステートでのパスメトリックの値をそれぞれ記憶するパスメトリック記憶手段と、選択したパスの出力を記憶するパスメモリと、最尤のパスメトリックのアドレスを検出しパスメモリの制御を行う最尤判定手段とから構成される。

【0009】ここで、上記ACS演算手段においては、いわゆるパスメトリックトランジションダイアグラムに従って、各ステートでの生き残りパスが選択され、この生き残りパスのパスメトリックが計算される。このパスメトリックトランジションダイアグラムは、上記図7に示すようなトレリス線図を基にして作られる。

【0010】上記図7に示すようなトレリス線図で示される符号が用いられている場合には、図8のA、Bに示すようなパスメトリックトランジションダイアグラムとなる。すなわち、図7に示すトレリス線図の場合、状態(00)で合流するのは、状態(00)から符号(00)を出力して生じるパスと、状態(10)から符号(11)を生じるパスとの2通りである。従って、今回の状態(00)のパスメトリック $S_{00}(\text{new})$ は、前時刻のステートの値(パスメトリック)を $S_{00}(\text{old})$ 、 $S_{10}(\text{old})$ 、ブランチメトリックを $BM_{00}$ 、 $BM_{11}$ のように表すとき、

$$S_{00}(\text{new})_a = S_{00}(\text{old}) + BM_{00}$$

$$S00(new)b = S10(old) + BM11$$

のいずれかとなる。すなわち、計算途中の2つのパスメトリック $S00(new)a$ 、 $S00(new)b$ の内の小さい方のパスが選択され、選択されたパスの値（パスメトリック）が今回の状態（00）のパスメトリック $S00(new)$ として出力されると共に、このときのパス選択情報も出力される。

【0011】また、状態（01）で合流するのは、状態（00）から符号（11）を出力して生じるパスと、状態（10）から符号（00）を生じるパスとの2通りである。従って、今回のパスメトリック $S01(new)$ は、  
 $S01(new)a = S00(old) + BM11$   
 $S01(new)b = S10(old) + BM00$   
 のいずれかとなる。また、状態（10）で合流するのは、状態（01）から符号（01）を出力して生じるパスと、状態（11）から符号（10）を生じるパスとの2通りである。従って、今回のパスメトリック $S10(new)$ は、

$$S10(new)a = S01(old) + BM01$$

$$S10(new)b = S11(old) + BM10$$

のいずれかとなる。また、状態（11）で合流するのは、状態（01）から符号（10）を出力して生じるパスと、状態（11）から符号（01）を生じるパスとの2通りである。従って、今回のパスメトリック $S11(new)$ は、

$$S11(new)a = S01(old) + BM10$$

$$S11(new)b = S11(old) + BM01$$

のいずれかとなる。このことに基づいて、図8のA、Bに示すようなパスメトリックトランジションダイアグラムを作ることができる。

#### 【0012】

【発明が解決しようとする課題】ところで、従来のビタビ復号器において、ACS演算を時分割処理で行い、かつACS演算の結果から得られたパス選択情報を記憶するパス選択情報メモリにRAM(Random Access Memory)を使用するものが知られている。このようなビタビ復号器では、ACS演算の結果から得られたパス選択情報を、単純に、RAMの1アドレス当たりのビット数に相当する情報が揃う毎にRAMに書き込み、全てのACS演算が終了してパス選択情報がRAMに書き込まれた後に、最尤判定のために過去のパス選択情報を順次読み出して、復号結果を推定して作成している。

【0013】この場合、時分割処理によるACS演算の結果得られたパス選択情報を全て書き込んだ後に過去のパス選択情報の読み出しを行うため、処理時間として書き込みと読み出しにそれぞれ別の時間が必要となり、例えばフリップフロップを使用したパス選択情報記憶回路に比べて処理時間が多く必要であり、ビタビ復号の高速化が実現できないという欠点がある。また、フリップフロップを使用したパス選択情報記憶回路の場合には、最

尤判定を行うための過去のステートのパス選択情報を読み出すときに全ステートの内容をアクセスしてしまうため、低消費電力化の障害となっている。

【0014】本発明は、上述した実情に鑑みてなされたものであり、一般のRAM等のメモリをパス選択情報メモリに用いながら、過去のパス選択情報の書き込み、読み出し処理の効率を高めることができ、低消費電力化やビタビ復号の高速化が可能なビタビ復号器の提供を目的とするものである。

#### 【0015】

【課題を解決するための手段】本発明は、ビタビ復号の際のACS演算を時分割処理で行い、ACS演算手段からのパス選択情報を複数のnステート分毎にまとめてパス選択情報記憶手段に記憶するようにし、nステート分のACS演算処理中のパス選択情報記憶手段へのパス選択情報の書き込み時以外の時間に最尤判定のための過去のパス選択情報を読み出すことにより、上述の課題を解決する。

【0016】ここで、上記ACS演算されて得られたパス選択情報を一時記憶手段に一時記憶することにより上記nステート分をまとめて上記パス選択情報記憶手段に送ることが好ましい。

#### 【0017】

【発明の実施の形態】以下、本発明に係る好ましい実施の形態について、図面を参照しながら説明する。

【0018】図1は、本発明の実施の形態が適用されるビタビ復号器の概略構成を示すブロック図である。

【0019】一般にビタビ復号器は、ビタビアルゴリズムを利用して、各状態での2つのパスの内の最尤（最も確からしい）パスを選択し、所定長まで生き残りパスの選択を行い、各状態で選択したパスの内の最尤のものを検出することで、受信符号を復号するものである。

【0020】このようなビタビアルゴリズムに基づいて畳み込み符号を計算するビタビ復号器は、基本的には図1に示すように、端子11からの受信データの系列（受信系列）と前述した各ブランチとの間のメトリックを計算するブランチメトリック演算部12と、生き残りパスを選択して生き残りパスのパスメトリックを計算するACS（アダプター、コンパレータ、セパレータ）演算部13と、各ステート（状態）でのパスメトリックの値をそれぞれ記憶するパスメトリック記憶部14と、選択したパスの選択情報であるパスの出力を記憶するパス選択情報記憶部15と、各時刻で最尤のパスメトリック値であったステートを検出する最尤検出部16と、このステート情報及びパス選択情報記憶部15から読み出した過去のパス選択情報を使用してデータ推定を行い復号データを端子18より出力するデータ推定部17とを有して構成されている。

【0021】ここで、畳み込み符号／ビタビ復号の具体例として、拘束長 $K=9$ 、符号化率 $R=1/2$ の場合に

ついて説明する。拘束長 $K=9$ の場合のステート（状態）数は、 $2^{K-1}$  となり、 $K=9$  のとき  $2^8 = 256$  ステートとなる。これらの256ステートを、2桁の16進数を用いて00～FFと表し、ステートxxの値であるパスメトリックをSxxのように表す。また、新たなステートのパスメトリックをSxx(new)、その前時刻のステートのパスメトリックをSxx(old)と表す。

【0022】上記拘束長 $K=9$ 、符号化率 $R=1/2$ の場合に、ACS演算においては、各状態（ステート）に遷移する2つのパスの内、最尤のパスを選択する演算を行う。例えば、新たなステート00(new)に遷移するパスは、前時刻のステート00(old)からと80(old)からの2つのパスであり、この前時刻のステート00(old)、80(old)からステート00(new)に遷移するためには、入力符号がそれぞれ、(00)、(11)である必要がある。このことから、入力符号（復調データ）からそれぞれの入力符号パターン（メトリック（ブランチメトリック）を求め、前時刻のメトリック（パスメトリック）と組み合わせて、パスの尤度を求め、生き残りパスの選択を行うと共に次の時刻のためのパスメトリックを記憶して保持する。

【0023】このときの計算は、入力符号パターンが( $G_0 G_1$ )のときのブランチメトリックをBM( $G_0 G_1$ )と表すとき、

$$S00(new)a = S00(old) + BM(00)$$

$$S00(new)b = S80(old) + BM(11)$$

if ( $S00(new)a < S00(new)b$ )

$$S00(new) = S00(new)a$$

else

$$S00(new) = S00(new)b$$

となり、このとき、 $S00(new)a$ 、 $S00(new)b$ のいずれを選んだかがパス選択情報となる。1時刻の間に、これをS00～SFFについての256ステート分行う。

【0024】ここで、本実施の形態においては、上記ACS演算部13でのACS演算を時分割処理で行うと共に、ACS演算により得られたパス選択情報を記憶するパス選択情報記憶部15として、複数nステート分のパス選択情報を1記憶単位として記憶するような1アドレス当たりnビットのRAM(Random Access Memory)を用い、時分割処理のACS演算を行って作成したパス選択情報をnステート分毎にRAMに書き込むようにしている。また最尤判定のパスのトレースバックのために使用する過去のパスの選択情報は、上記nステートのACS演算処理中に、パス選択情報記憶部15のRAMから上記書き込み以外の時間に割り当てて読み出す。

【0025】具体的には、例えば図2に示すように、2つのACS演算回路13a、13bにてそれぞれ128多重の時分割処理により上記256ステート分のACS演算を行い、このACS演算の結果から得られたパス選択情報を、一時記憶回路20に送って一時的に保持し、

nステート分、例えば8ステート分のパス選択情報をまとめてパス選択情報記憶部15に送って書き込むと共に、この書き込み時以外の処理タイミングでパス選択情報記憶部15から過去のパス選択情報を読み出して、上記1時刻の間に書き込みと読み出しとが交互に繰り返行われるようにしている。

【0026】この図2において、上述した復調データである符号( $G_0 G_1$ )が所定周期Tで入力される毎にブランチメトリック演算部12で作成したブランチメトリックBM( $G_0 G_1$ )を、ACS演算回路13a、13bにそれぞれ入力する。ACS演算回路13a、13bでは、256ステート分のACS演算の内の、偶数128ステート分のパスメトリック(S00、S02、・・・、SFE)の演算をACS演算回路13aで行い、奇数128ステート分のパスメトリック(S01、S03、・・・、SFF)の演算をACS演算回路13bで行っている。このACS演算においては、上記128多重時分割によりT/128毎に処理を行っており、パスメトリックメモリ14から1時刻前(1T前)のパスメトリックを順次読み出して、それぞれの処理毎にパス選択情報を作成し、一時記憶回路20を介してパス選択情報記憶部15に書き込むと共に、新しいパスメトリックをパスメトリックメモリ14に書き込んでいる。

【0027】また、ACS演算回路13a、13bで毎回の処理毎に作成した新しいパスメトリックを最尤検出部16に送り、全てのステートのACS演算が終了した時点で、最尤のステートを検出し、データ推定をするための過去のパス選択情報を選択するために、パス選択情報記憶部15の初期アドレス値を作成する。メモリ読出/書込アドレス制御部21では、上記初期アドレス値を入力し、パス選択情報記憶部15のアドレスを作成する。書き込み時以外の残りの時間（例えば4処理に3回）に、パス選択情報記憶部15より1時刻前のパス選択情報を読み出し、メモリ読出/書込アドレス制御部21に送り、さらに1時刻前のパス選択情報のアドレスを作成する。パス選択情報記憶部15は例えば64時刻(64T)分のパス選択情報を蓄えるようになっており、このパス選択情報記憶部15からのパス選択情報をデータ推定部17に送る。データ推定部17では、これらの過去のパス選択情報と、最尤検出部16からの出力とを用いてデータ推定を行い、復号データを端子18より出力する。

【0028】ここで、パス選択情報の読出/書込動作について、図3を参照しながら説明する。

【0029】この図3において、上記ACS演算の256ステート分の処理の周期Tは、復号データの更新周期であり、この周期Tを128多重で時分割して各ACS演算回路13a、13bによる処理をそれぞれ行わせている。すなわち、図3のAは偶数ステートのACS演算部13aからのパス選択情報P0、P2、・・・、P2

54の出力を示し、Bは奇数ステートのACS演算部13bからのパス選択情報P1、P3、・・・、P255の出力を示している。これらのACS演算は、上記周期Tを128分割した各タイムスロット( $T/256$ )毎にそれぞれ毎行われ、新たなパス選択情報がそれぞれ出力される。各パス選択情報P0～P255は、前述したように、各状態(ステート)での2つのパスの内の最尤のパスを選択するそれぞれ1ビットの情報であり、1時刻周期Tの間には256ステート分の256ビットのパス選択情報が得られる。

【0030】一時記憶回路20は、上記 $T/128$ のタイムスロット毎に各ACS演算部13a、13bからそれぞれ得られるパス選択情報について、4スロット分( $4T/128=T/32$ )まとめて、計8ステート分の8ビットのパス選択情報を単位としてパス選択情報記憶部15に送っている。すなわち、図3のCは偶数ステートのACS演算回路13aの上記4スロットの内の最初のスロットのパス選択情報(例えばP0)を取り込んで保持して得られる出力を示し、以下同様にDは上記4スロットの内の2番目のスロットのパス選択情報(例えばP2)を一時保持した出力を示し、Eは上記4スロットの内の3番目のスロットのパス選択情報(例えばP4)を一時保持した出力を示している。図3のFは、上記Aのパス選択情報出力そのものであり、図3のHに示すように、上記4スロットの内の最後の4番目のスロットでパス選択情報記憶部15の書き込みを行うことに対応して、この書き込みタイミングで偶数ステートのACS演算回路13aからの4スロット分のパス選択情報(例えばP0、P2、P4、P6)がそろることになる。奇数ステートのACS演算回路13bからのパス選択情報(図3のB)についても同様にして、4スロット分のパス選択情報が上記パス選択情報記憶部15の書き込みタイミングでまとめられ、上記偶数ステートの4スロット分と合わせて、計8ステートの8ビット分のパス選択情報が1つの記憶単位としてパス選択情報記憶部15に書き込まれる。なお、上記4スロット毎の4番目のスロットのパス選択情報については、各ACS演算回路13a、13bからのパス選択情報出力をそのまま図3のF、Gのように用いればよいことから、一時記憶回路20としては、6ビットのラッチで構成することができる。

【0031】パス選択情報記憶部15としては、1アドレスが8ビット(1バイト)で、1時刻(1T)当たり全256( $=8 \times 32$ )ステートを64時刻分記憶できるような記憶容量のRAMを用いている。すなわち、8ビット $\times$ (32 $\times$ 64)アドレス=2048バイト(2Kバイト)の記憶容量となる。

【0032】パス選択情報記憶部15からの読み出しは、図3のHに示すように、上記書き込み時以外の残りの時間、すなわち上記4スロットの内の3スロット(3

$T/128$ )で行うようにすればよい。具体的には、上記最尤検出回路16により検出された最尤のステートに基づいて、メモリ読出/書込アドレス制御部21がパス選択情報記憶部15の読み出しアドレスRAを作成し、パス選択情報記憶部15から1時刻前の最尤ステートのパス選択情報を読み出す。上記最尤ステートのパス選択情報は1ビットであるが、この1ビットを含む同じアドレスの8ビット(8ステート分)のパス選択情報P

(1)を読み出す。この同時に読み出された8ビットのパス選択情報P(1)中の上記最尤ステートの1ビットに基づいて、メモリ読出/書込アドレス制御部21により、さらに1時刻前の最尤ステートのパス選択情報の1ビットを含む8ビットを読み出すためのアドレスを作成する。これを繰り返して、上記64時刻前のパス選択情報までを読み出したときに、データ推定部17によりデータを推定し、復号データを端子18より出力する。

【0033】図3のIは、メモリ読出/書込アドレス制御部21による読み出しアドレスRA及び書き込みアドレスWAの生成タイミングを示している。また、図3のJは、パス選択情報記憶部15から読み出された過去のパス選択情報をP(i)として表しており、P(i)は、過去のi時刻前の最尤ステートのパス選択情報の1ビットを含む同じアドレスの8ビットのデータを示している。

【0034】以上の具体例においては、上記128多重時分割のACS演算により、1時刻(周期T)が128のタイムスロットに分割されて各タイムスロット毎に偶数及び奇数ステートのACS演算処理が行われており、8ステート分のパス選択情報を出力するための4タイムスロットの内の1スロットでパス選択情報の書き込みを行うと共に残りの3スロットで過去のパス選択情報の読み出しを行っている。読み出したパス選択情報の選択においては、8ステート毎に8ビットの情報単位でアドレスを構成し書き込んでいるため、過去の各時刻でのパス選択情報の読み出しの際に、上記具体例の場合には、全ステートの1/32の8ビットのパス選択情報を読み出し選択するだけでよい。ただし、パス選択情報記憶部15は過去64時刻(64T)分のパス選択情報を記憶する容量を有しているため、1時刻Tの128のタイムスロットの内、32スロットをパス選択情報の書き込みに用い、64スロットを過去64時刻分の読み出しに用いることになる。このように1時刻Tの間に読み出しと書き込みとを交互に繰り返すことにより、全256ステート分のパス選択情報を書き込んで、過去の全64時刻分のパス選択情報を読み出すことが、1時刻Tの間に行える。従って、RAM等のメモリをパス選択情報記憶部15に用いながら、ビタビ復号の動作速度を低下させることなく、低消費電力化が実現できる。

【0035】なお、上述の例においては、1時刻(周期

T)の1/128のタイムスロットの内、32スロットをバス選択情報の書き込みに用い、64スロットを過去64時刻分の読み出しに用いているが、過去のバス選択情報の記憶容量を増やして、過去96時刻分をパストレースに用いるようにすることもできる。

【0036】以上説明したようなビタビ復号器が用いられる機器の一例について、図4及び図5を参照しながら説明する。

【0037】図4は、上述したようなビタビ復号器を用いて誤り訂正を行うチャンネルコーデック（符号復号器）を有するディジタル移動体通信機の端末の概略構成を示している。

【0038】この図4に示すディジタル移動体通信機の端末は、ベース局とのキャリア信号の入出力をアンテナ100で行い、送受信のキャリア信号を送受共用器101を介して、受信機102で復調処理を、送信機104で変調処理を行う。シンセサイザ103は、受信機102、送信機104の送受信周波数を制御する。

【0039】受信機102からの出力信号は、チャンネルデコーダ105に送られ、このチャンネルデコーダ105では、図5に示すビタビ復号器202を含み、音声及び通信制御情報のフォーマット変換及び復号が行われる。チャンネルデコーダ105からの音声データは、音声コーデック107に送られる。音声コーデック107では音声の符号/復号処理が行われ、音声データがチャンネルエンコーダ106に送られて、音声及び通信制御情報のフォーマット変換及び符号化処理が行われる。チャンネルデコーダ105とチャンネルエンコーダ106とで、いわゆるチャンネルコーデック200が構成される。

【0040】音声コーデック107には送受話器108が接続されており、この送受話器108でスピーカ111、マイクロホン112とのインターフェース変換を行い、音声通話を実現する。

【0041】また、コントローラ113では、制御情報の作成、解読を行い、呼の設定、解除、維持を行うと共に、キー/ディスプレイ109等のI/O制御を行う。送受信周波数の制御を行うシンセサイザ103の制御も、コントローラ113により行われる。

【0042】図5は、上記チャンネルコーデック200の具体的構成及びその周辺を示すブロック図であり、ビタビ復号器202に上述したような本発明の実施の形態のビタビ復号器が用いられるわけである。

【0043】この図5において、音声コーデック107からの音声データやコントローラ113からの制御情報は、チャンネルエンコーダ106のCRCジェネレータ204に送られてCRCが付加され、畳み込み符号器205で誤り訂正用の符号化が行われた後に、インターリーブ206でインターリーブ処理が施されて、変調の前処理までを行い、送信機104に送っている。

【0044】受信機102から出力される受信データ

は、チャンネルデコーダ105のデインターリーブ201でインターリーブ処理が施され、上述した実施の形態のようなビタビ復号器202に送られて誤り訂正処理が施された後、CRCチェッカ203に送られてデータの正誤が確認される。CRCチェッカ203からの音声データは音声コーデック107に送られ、制御データはコントローラ113に送られる。

【0045】なお、本発明は上述した実施の形態のみに限定されるものではなく、例えば、上述した具体例においては、1時刻（周期T）の1/128のタイムスロットの内、32スロットをバス選択情報の書き込みに用い、64スロットを過去64時刻分の読み出しに用いているが、過去のバス選択情報の記憶容量を増やして、過去96時刻分をパストレースに用いるようにすることもできる。また、上記時分割多重の多重化の数は任意に設定でき、一時記憶回路のビット数やバス選択情報記憶部のメモリの1アドレスあたりに記憶するバス選択情報の数も任意に変更することができる。また、符号の拘束長、符号化率等も任意に設定できる。さらに、本発明は、チャンネルコーデックを有するディジタル移動体通信機の他にも、種々のビタビ復号器を用いる機器に適用できることは勿論である。

【0046】

【発明の効果】本発明によれば、ビタビ復号器のACS演算を時分割処理で行い、バス選択情報記憶手段に複数のnステート分のバス選択情報をまとめて記憶させ、nステート分のACS演算を行っている間の書き込み時以外の時間で過去のバス選択情報を読み出しているため、バス選択情報記憶手段に通常のRAM等のメモリを用いながら、ACS演算の1処理あたりに必要とされるメモリアクセスの回数を低下させることができ、ビタビ復号動作を高速化できる。また、バス選択情報の書き込みをnステート単位で行うことにより、メモリの1アドレスあたりの書き込み量をnビットとすることができ、バス選択情報の読み出しの際に、全てのステートのバス選択情報を読み出す必要がなく、低消費電力化に有効である。

【図面の簡単な説明】

【図1】 本発明の実施の形態が適用されるビタビ復号器の概略構成を示すブロック図である。

【図2】 図1のビタビ復号器のバスメトリックメモリ近傍の具体的な構成の一例を示すブロック図である。

【図3】 ACS演算及びバス選択情報記憶部のアクセス動作の具体例を説明するためのタイムチャートである。

【図4】 ディジタル移動体通信機の概略構成を示すブロック図である。

【図5】 ビタビ復調器が用いられるチャンネルコーデック及びその近傍の構成例を示すブロック図である。

【図6】 畳み込み符号の符号器の一例を示すブロック

図である。

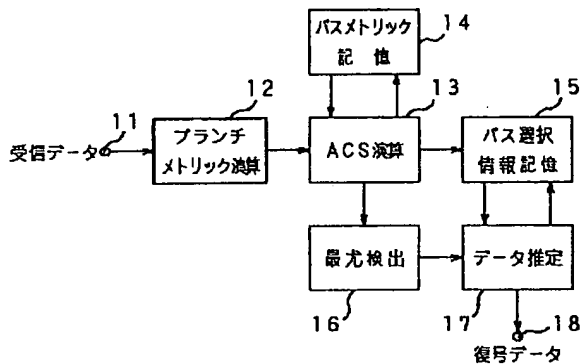
【図7】 ビタビ復号を説明するためのトレリス線図を示す図である。

【図8】 ビタビ復号を説明するためのパスメトリックトランジションを示す図である。

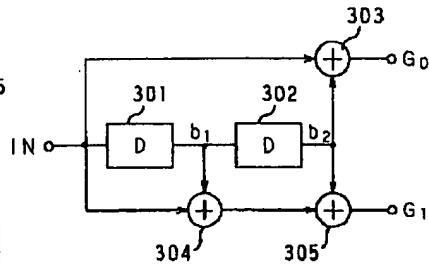
【符号の説明】

12 ブランチメトリック演算部、 13 ACS演算部、 13a, 13b ACS演算回路 14 パスメトリック記憶部、 15 パス選択情報記憶部、 16 最尤検出部、 17 データ推定部、 20 一時記憶回路、 メモリ読出/書込アドレス制御部

【図1】



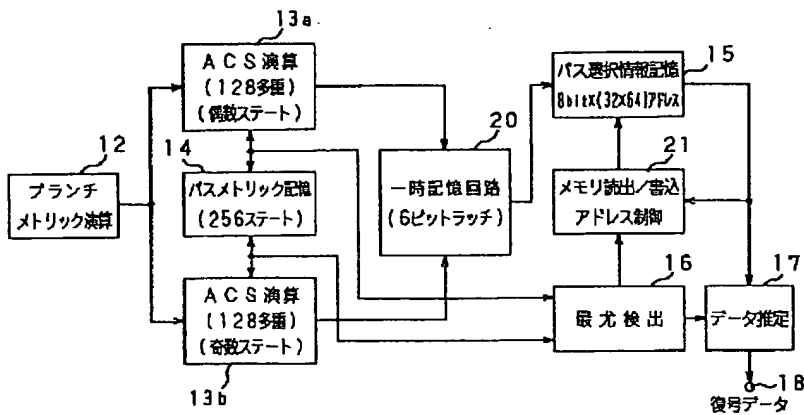
【図6】



畳み込み符号器の一例を示す図

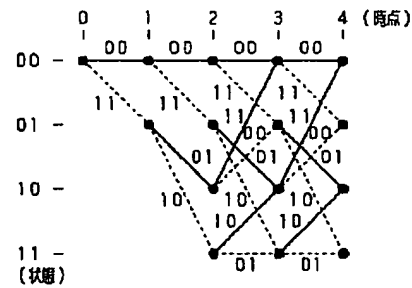
ビタビ復号器の概略構成を示す図

【図2】



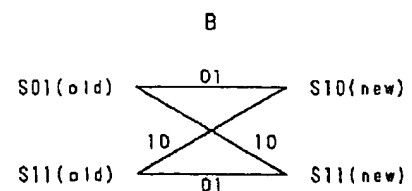
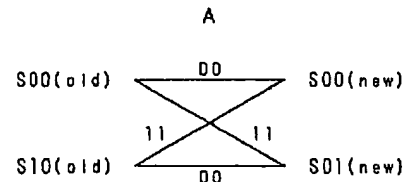
パスメトリックメモリ近傍の構成例を示す図

【図7】



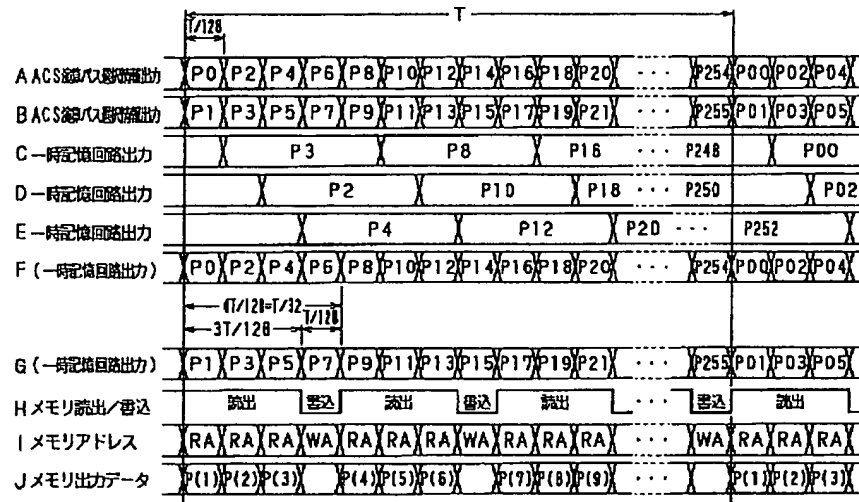
ビタビ復号のトレリス線図

【図8】



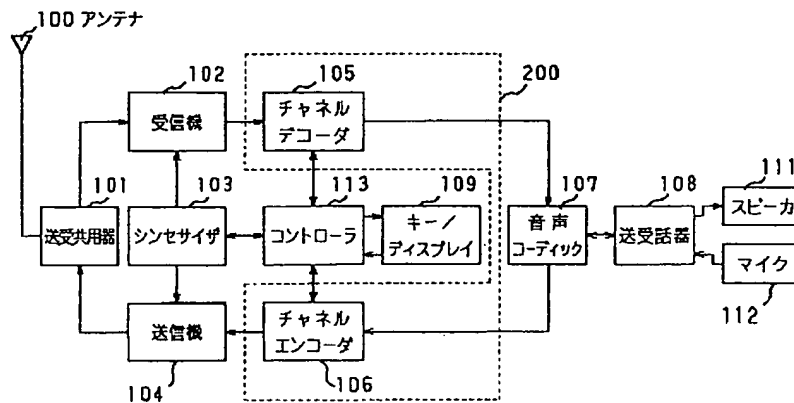
パスメトリックトランジションを示す図

【図3】



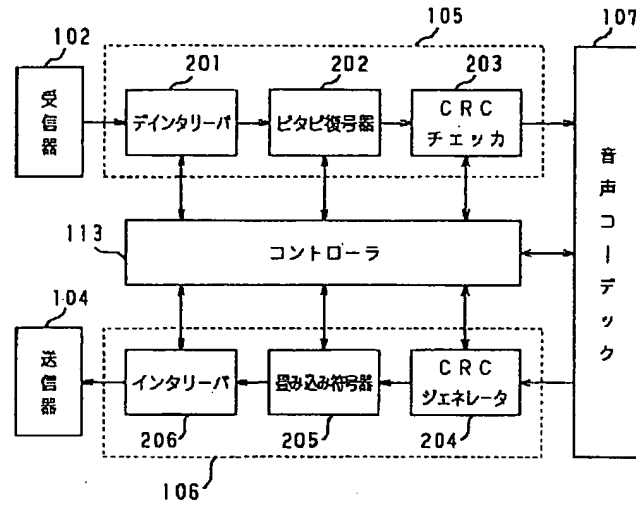
ACS演算及びバス選択情報メモリアクセス動作の具体例を示す図

【図4】



ディジタル移動体通信機の回路構成を示すブロック図

【図5】



チャネルコーデック及びその近傍を示すブロック図